# R Basic

- Assign value : "=" or "<-"
- sqrt(x) : $\sqrt{x}$
  log(x) : $\ln x$
  exp(x) : $e^x$
  abs(x) : $|x|$
- rep(x, times=y) : Repeat x for y times
- * Use # for comments.
- ls() : show memory
  rm() : remove variables
- c($x_1, x_2 \dots x_n$) : Create a vector
  (c: concatenate)
- 2:7 : Create a sequence 2,3,...7
  or
  seq(from=2, to=7, by=1)

- matrix(c($x_1, x_2 \dots x_n$), nrow = k, byrow = TRUE / FALSE)

$$\Rightarrow \begin{bmatrix} x_1 & x_2 & \cdots & x_k \\ x_{k+1} & \cdots & \cdots & x_{2k} \\ \vdots & & & \vdots \\ x_{n-k+1} & \cdots & \cdots & x_n \end{bmatrix} \quad \begin{bmatrix} x_1 & x_{k+1} & \cdots & x_{n-k+1} \\ x_2 & x_{k+2} & \cdots & \vdots \\ \vdots & \vdots & & \vdots \\ x_k & x_{2k} & \cdots & x_n \end{bmatrix}$$

* use help( ) or ? to get help

- read.csv( path, header = TRUE) : open csv file (header = TRUE: The first row are variable names)
  file.choose( ) : popping up menu
  read.table( path, header = TRUE, sep = ",")
  read.delim( path, header = TRUE) : open tab-delimited text file

  * Use readx! package to import .xlsx & .xls

- Export data using "write.table()"

  write.table( dataToExport, file = "ExportedFileName.csv", row.names = F, sep = ",")

  use absolute path if want to save the file outside current working directory

  write.csv( )
  write.csv2( )

# Getting Started with Data in R

help(command)
?command : help menu

* file.choose() is useful to replace file path
* Using "Import Data" menu in R, we can even import data without coding!
* Data$Variable
or
attach(Data), detach(data)
Both can be used to get data from dataset
* Data can be overwritten after attached
* cbind(): To combine data to dataset in a column-wise way.
* getwd(): getting current working directory
setwd("path"): setting working directory
* save.image(): Save workspace image
load("filename"): load workspace image

* read.table(file="path", header=TRUE, sep=",")
= read.csv when sep=","

dim(data): How many rows
How many column

* head(data)
tail(data): Getting the first or last six data

name(data): Getting column names

* class(): Determine
level(): class of data
* as.factor(): Save data as factorial

* Summary(): Do summary of the dataset